

APPENDIX VIII - REBOOT EVENT

© Copyright 2003 Time Warner Cable, Inc. All rights reserved.

```
import java.lang.*;
5 public class RebootEvent implements IMessageEvent
{
    public final static int REBOOT_BUTTON_PRESSED = 0x30000001;
    public final static int REBOOT_BY_IMPLEMENTATION = 0x30000002;
    public final static int REBOOT_BY_TRUSTED_APP = 0x30000003;
10 public final static int REBOOT_FOR_UNHANDLED_SYS_ERROR = 0x30000004;
    public final static int REBOOT_FOR_UNHANDLED_APP_ERROR = 0x30000005;

    // Static reboot event Id number. Incremented for each new event.
15 private static long lastEventId;

    private long eventId;
    private long appId;
    private int typeCode;
    private long date;
20 private String message;

    public RebootEvent(int typeCodeIn, long appIdIn, String messageIn)
    {
        lastEventId = (lastEventId == Long.MAX_VALUE) ? 0 : lastEventId + 1;
25 eventId = lastEventId;
        typeCode = typeCodeIn;
        appId = appIdIn;
        date = System.currentTimeMillis();
        message = messageIn;
30 }

    /**
     * Get the unique identifier of the event.
     *
35 * @return unique identifier of the event.
     */
    public long getEventId()
    {
        return eventId;
40 }

    /**
     * Get the unique identifier of the application logging the event.
     *
45 * @return The application identifier.
     */
    public long getAppId()
    {
        return eventId;
50 }

    /**
     * Get the event type code. Identifies a code specific to the event system.
     *
55 * @return type code of the event.
     */
    public int getTypeCode()
```

```

    {
        return typeCode;
    }

5  /**
   * Set the error type code. Allows an application to change the error received.
   *
   * @param error - The new error value.
   */
10 public void setTypeCode(int typeCodeIn)
    {
        typeCode = typeCodeIn;
    }

15 /**
   * Get the event date in milli-seconds from midnight January 1, 1970.
   *
   * @return Date the event was submitted to the system.
   */
20 public long getDate()
    {
        return date;
    }

25 /**
   * Set the date. Allows an application to change the date received.
   *
   * @param error - The new error value.
   */
30 public void setDate(long dateIn)
    {
        date = dateIn;
    }

35 /**
   * Get the readable message.
   *
   * @return the readable event message.
   */
40 public String getMessage()
    {
        return message;
    }
    protected void finalize()
45 {
        // Store the lastRebootEventId in a persistent storage object.
    }
    static {
        if(/*some persistent storage object exists with reboot event Id stored in it*/ true)
50     lastEventId = /* value in the persistent storage object */1;
        else
            lastEventId = 0;
    }

55 }

```